

Nome: Pedro Henrique Bertotto
Ciência da Computação

CRISE DO SOFTWARE

indústria de software, especialmente em relação à qualidade, custos e prazos. Essa crise tem sido observada desde a década de 1960, quando o desenvolvimento de software começou a se tornar uma prática comum em empresas e instituições.

Um dos principais fatores que contribuem para a crise do software é a complexidade dos sistemas desenvolvidos. À medida que os softwares se tornam cada vez mais complexos, a sua qualidade e confiabilidade tornam-se mais difíceis de serem garantidas. Além disso, muitas empresas ainda utilizam métodos de desenvolvimento de software desatualizados, que não levam em consideração os avanços tecnológicos e as necessidades dos usuários finais.

Outro fator que contribui para a crise do software é a falta de comunicação e colaboração entre as equipes envolvidas no desenvolvimento de um sistema. Muitas vezes, as equipes responsáveis por diferentes partes do software trabalham de forma isolada, sem compartilhar informações ou coordenar seus esforços. Isso pode levar a problemas de integração e atrasos no desenvolvimento do sistema como um todo.

Além disso, a pressão por redução de custos e prazos também contribui para a crise do software. Muitas empresas não investem o suficiente em recursos humanos e tecnológicos para garantir a qualidade e confiabilidade dos softwares desenvolvidos, o que pode resultar em problemas graves após a implantação do sistema.

Para enfrentar a crise do software, as empresas devem adotar métodos de desenvolvimento de software mais modernos e eficazes, que levem em consideração as necessidades dos usuários finais e as melhores práticas de desenvolvimento. Além disso, é importante investir em recursos humanos e tecnológicos de alta qualidade, para garantir a qualidade e confiabilidade dos softwares desenvolvidos.

Em resumo, a crise do software é um desafio significativo para a indústria de software, mas pode ser enfrentada com a adoção de métodos de desenvolvimento mais modernos e eficazes, e o investimento em recursos humanos e tecnológicos de alta qualidade. Com essas medidas, a qualidade e confiabilidade dos softwares desenvolvidos podem ser significativamente melhoradas, e os custos e prazos podem ser reduzidos.

REQUISITOS FUNCIONAIS, NÃO FUNCIONAIS E DE SISTEMA

Os requisitos de software são as especificações que definem o que o software deve fazer e como ele deve se comportar. Existem dois tipos principais de requisitos: funcionais e não funcionais.

Os requisitos funcionais descrevem o que o software deve fazer. Eles são as funções e recursos que o software deve ter para atender às necessidades do usuário. Por exemplo, um software de edição de texto precisa ter recursos como a possibilidade de inserir, excluir e formatar texto, bem como a capacidade de salvar e abrir arquivos.

Já os requisitos não funcionais definem como o software deve se comportar. Eles são os critérios de qualidade que o software deve atender, tais como desempenho, segurança, confiabilidade e usabilidade. Por exemplo, um software de edição de texto precisa ser fácil de usar, rápido e seguro, para que o usuário possa trabalhar com eficiência e sem riscos de perda de dados.

Além disso, existem os requisitos de sistema, que definem as características do ambiente em que o software será executado. Eles incluem requisitos de hardware, como a capacidade de processamento e memória, requisitos de sistema operacional e outros requisitos específicos do ambiente em que o software será utilizado.

É importante que os requisitos funcionais, não funcionais e de sistema sejam cuidadosamente definidos e documentados, para que todos os membros da equipe de desenvolvimento tenham uma compreensão clara do que deve ser desenvolvido e para que o software seja construído de acordo com as expectativas do usuário.

Os requisitos também devem ser validados para garantir que o software atenda às expectativas do usuário. Isso pode ser feito por meio de testes de aceitação, que verificam se o software cumpre todos os requisitos definidos.

Em resumo, os requisitos funcionais definem o que o software deve fazer, os requisitos não funcionais definem como o software deve se comportar e os requisitos de sistema definem as características do ambiente em que o software será executado. É importante que todos esses requisitos sejam cuidadosamente definidos e validados para garantir que o software atenda às expectativas do usuá